



## THATCamp Paris 2012 Non-actes de la non-conférence des humanités numériques

Éditions de la Maison des sciences de l'homme

---

# L'historien programmeur ?

Proposé par : Frédéric Clavert, Aurélien Berra, Franziska Heimburger

Collectif

---

DOI : 10.4000/books.editionsmsmh.305

Éditeur : Éditions de la Maison des sciences de l'homme

Lieu d'édition : Paris

Année d'édition : 2012

Date de mise en ligne : 1 octobre 2012

Collection : La Non-Collection

ISBN électronique : 9782735115273



<http://books.openedition.org>

### Référence électronique

COLLECTIF. *L'historien programmeur ?* Proposé par : Frédéric Clavert, Aurélien Berra, Franziska Heimburger

In : *THATCamp Paris 2012 : Non-actes de la non-conférence des humanités numériques* [en ligne]. Paris :

Éditions de la Maison des sciences de l'homme, 2012 (généré le 01 mai 2019). Disponible sur

Internet : <<http://books.openedition.org/editionsmsmh/305>>. ISBN : 9782735115273. DOI : 10.4000/books.editionsmsmh.305.

---

Ce document a été généré automatiquement le 1 mai 2019.

---

# L'historien programmeur ?

Proposé par : Frédéric Clavert, Aurélien Berra, Franziska Heimburger

Collectif

---

## Introduction

- 1 Entendons ici, le terme d'historien au sens large. En tant que chercheurs en sciences humaines et sociales, nous manions tous des documents, des sources primaires. Il ne s'agit pas de poser la question de la programmation par les chercheurs eux-mêmes, mais celle de la relation entre le programmeur et l'historien en partant d'exemples concrets pour ensuite en discuter<sup>1</sup>. Nous avons pris, comme point de départ, les deux éditions de *Programming Historian*<sup>2</sup>.
- 2 Trois éléments doivent être abordés : pourquoi sommes-nous amenés à nous poser la question du code ? Cette interrogation débouche sur une autre : quelle est la place du chercheur dans la chaîne d'un projet scientifique impliquant du développement ?
- 3 Dans quels cas est-il nécessaire de se lancer dans un peu « ou beaucoup » de programmation ?
- 4 Y a-t-il des exemples concrets où une programmation par un historien a été décisive dans sa recherche ?

## I. Pourquoi un chercheur se pose la question du code ?

3

- 5 Pour la première édition de *The Programming Historian*, les deux auteurs avaient des profils différents : un informaticien converti à l'histoire avec 30 ans d'expérience du code informatique (William J. Turkel) et un historien ayant appris récemment à développer pour les besoins d'un projet précis (Alan McEachern). Pour la seconde édition, l'équipe, plus large et d'autant plus hétérogène, donne un très bon exemple de l'imbrication de nombreux corps de métiers et savoir-faire articulés autour d'une plate-forme

collaborative et typique des humanités numériques. Cette édition est plus variée dans les sujets qu'elle aborde et les outils et langages de programmations utilisés<sup>4</sup>.

- 6 Arrêtons-nous, un instant, sur la première version.
- 7 Le but des deux auteurs était de montrer comment récupérer de l'information sur le web, comment la stocker dans Zotero, pour ensuite exploiter les données récupérées. C'est ce que l'on appelle du *data mining*. Cette méthode a été appliquée aux archives en ligne *The Old Bailey on-line*<sup>5</sup> et résumée dans le très bon rapport « Data mining with criminal intent<sup>6</sup> ». L'origine de ce livre repose sur la conviction des deux auteurs que l'histoire numérique est le futur de la discipline historique. L'histoire numérique suppose-t-elle toutefois d'acquérir les compétences nécessaires au développement informatique ?
- 8 Frédéric Clavert explique pourquoi il en est venu à s'intéresser à ces questions. *Il a eu besoin de trouver* sur le web, de stocker dans Zotero et de manipuler des données sur l'Union économique et monétaire. Il s'agit ici d'une programmation très légère, quelque chose que l'on peut qualifier de « script » plutôt que de « développement » à proprement parler.
- 9 Toutefois, les humanités numériques sont très orientées vers la pratique et la méthodologie, plus que vers la théorie. Comment appréhender ce qu'implique cette pratique sans avoir des notions de code ? Il s'agit là d'un autre aspect de l'historien programmeur : non apprendre à développer pour son petit projet, mais apprendre à développer pour comprendre les grands projets de recherche impliquant plusieurs corps de métiers, dont des programmeurs. Par exemple, il est difficile de saisir pleinement le cycle de développement au sein d'un projet si l'on ne sait pas écrire un peu de code. Ainsi peut-on réaliser la complexité du développement, qui doit prendre en compte tous les comportements possibles de l'utilisateur pour lequel on conçoit un programme.
- 10 On peut avoir des expériences inverses : mettre un corpus de documents variés en ligne et proposer des outils d'exploitation de ce corpus. Le chercheur ne rencontre pas d'obstacles sur certaines opérations, comme modifier des textes formatés en HTML. Dans ces cas, l'intervention directe du chercheur permet même un certain gain de temps. Mais il y a un problème de veille sur les nouveaux outils, comparaison des diverses applications disponibles, analyse des avantages/faiblesses des diverses possibilités en fonction des données spécifiques.
- 11 Le conseil et l'orientation sont cruciaux pour les choix de l'historien. Dans le secteur privé, on fait appel à des sociétés de consultants spécialisés dans divers domaines. Ces consultants travaillent sur des projets similaires et mutualisent les expériences et les connaissances. Les chercheurs en sciences humaines et sociales (SHS) manquent de ce genre de structure où des « consultants », informaticiens spécialisés en SHS, pourraient intervenir à des degrés et avec des implications diverses.
- 12 Une autre expérience a été partagée avec le langage d'interrogation de bases de données SQL. Comment traiter et stocker des données qu'on produit ? La prise de note doit-elle être faite avec Word ? En XML TEI ? La prise de notes étant collective, il est nécessaire de modéliser les données et il devient important de faire comprendre à une équipe complète que produire des données revient à produire un modèle de données. La modélisation des données est, en fait, la première dimension scientifique de l'entreprise. La seconde touche aux partages des données : chaque chercheur produisant des données dans un contexte de recherche précis doit apprendre à distinguer les questionnements scientifiques des données brutes. Passé la modélisation, on touche aux limites de ce que l'historien peut

faire (gestion du serveur, choix des logiciels, etc.). Mais la question des compétences techniques revient au moment de l'exploitation des données. Ensuite vient le point crucial : les chercheurs doivent connaître suffisamment le SQL pour pouvoir utiliser les données et produire de nouvelles connaissances.

- 13 Ainsi, besoin de différencier les technologies choisies du contenu et de son traitement. La pratique historique de l'informatique n'est pas la gestion du serveur, la sécurisation du système, le choix des technologies utilisées, mais il lui faut les compétences nécessaires pour exploiter la richesse des documents disponibles<sup>7</sup>.
- 14 Ainsi se pose la question de la place et du rôle du chercheur lorsqu'il élabore un projet nécessitant un développement informatique.

## II. La place du chercheur dans la chaîne de développement, de l'expression du besoin à l'aboutissement du projet

- 15 La discussion entre l'ensemble des participants de l'atelier s'est clairement orientée vers le rôle du chercheur au sein d'un projet de recherche impliquant du développement d'une part, et vers la nécessité de formation et de culture numériques des chercheurs d'autre part.

### A. La place du chercheur au sein d'un projet numérique

- 16 Évoquant un projet d'histoire médiévale, une participante a formulé un besoin spécifique. Elle s'est adressée à un informaticien qui a développé un plug-in qui répondait à ses besoins. L'expérience – une discussion entre historienne et développeur – a été riche. Avec surprise, elle a rapidement constaté que l'informaticien lui faisait part de nouvelles perspectives sur l'interprétation des manuscrits médiévaux. Elle a par la suite suivi une formation en Java pour mieux communiquer avec son informaticien<sup>8</sup>.
- 17 Dans ce dialogue entre programmeur et chercheur, il faut insister sur la modélisation des données, dès le début du projet. Ainsi, faut-il revenir sur la question de la modélisation et du risque à prendre lorsqu'au début d'un projet des choix sont faits, impliquant à titre principal le chercheur. On ne peut revenir sur ces décisions, très difficiles à prendre.
- 18 Le rôle du chercheur dans la modélisation des données repose sur l'explicitation de ce qu'il veut faire. Il n'est pas sain de programmer tout seul et de faire, ensuite, l'exploitation des données. L'autre rôle, très important, du chercheur est de faire comprendre aux développeurs ce qu'est le monde des sciences humaines et sociales. Dans le cadre d'un travail impliquant informaticiens et paléographes, ces derniers ont pris deux jours à expliquer que la paléographie ne fonctionne pas avec des réponses vrai/faux ou des pourcentages d'exactitude.
- 19 Le rôle du chercheur dans un projet numérique change la temporalité de la recherche. Pour un projet européen, les sociologues impliqués ont voulu donner accès à l'ensemble des interviews réalisées, plutôt que de laisser chaque groupe national collecter de son côté les données. Il a ainsi fallu un énorme travail en amont, avec les risques que cela comporte, pour modéliser et gérer les données. Pour être sûr de mener cet effort à bien, les chercheurs ont été obligés d'explicitier leurs buts, d'harmoniser le modèle de données,

ce qui implique des choix techniques, mais également scientifiques. Or, s'il faut rendre la recherche explicite dès le début d'un projet, sa temporalité est modifiée et la recherche, plus risquée.

- 20 La modification de la temporalité de la recherche pousse à se pencher sur la question de l'économie du temps. Selon l'un des historiens présents, la recherche est un processus long, qui part de documents, de leur analyse, engendrant un besoin de traitement informatique. À ce moment-là, intervient le dialogue entre informaticien et historien, puis le développement propre par l'informaticien. Enfin, l'historien doit développer la maîtrise de son nouvel outil de travail, s'y familiariser. Dans certains cas, on doit envisager qu'il soit plus rapide, plus efficace que l'historien développe lui-même son outil, dont la modélisation se fait dès la lecture de la source par l'historien. Jean-Philippe Genet parle à ce propos d'une « méta source »<sup>9</sup>. Seul l'historien est capable de construire cette méta source – il ne peut pas déléguer à l'informaticien cette tâche dans la mesure où l'enjeu épistémologique est majeur : c'est sa connaissance de la période et son travail d'analyse des sources qui vont lui permettre de concevoir un modèle de données pertinentes.
- 21 Programmer ou savoir utiliser les logiciels ? Au cours d'une expérience d'une base de données collaborative associée à un projet d'analyse d'images, les interfaces et la base de données ont été créées par l'informaticien, avec des contraintes d'ergonomie et après le conseil des historiens. Cela a bien fonctionné. La conclusion à en tirer est que les chercheurs doivent d'abord connaître les logiciels développés pour eux, pas nécessairement qu'ils sachent programmer. Des logiciels puissants existent, il n'est pas toujours nécessaire de réinventer la roue. L'ignorance informatique des historiens doit être évitée, mais il est nécessaire de disposer d'une bonne culture informatique, autorisant un bon apprentissage des logiciels.

## B. Formation et culture numériques des chercheurs

- 22 Puis vient la question de la formation et de la culture numériques des historiens. Ce qui semble important est qu'une culture informatique comprenant les bases de la programmation – mais pas nécessairement une pratique suivie et autonome – permette aux historiens – sociologues, anthropologues, – d'acquérir un vocabulaire les autorisant à dialoguer avec des informaticiens et à faire comprendre leurs besoins et leurs idées.
- 23 Quelle place doivent prendre ces éléments dans la formation des jeunes historiens ? Comment définit-on cette culture numérique ?<sup>10</sup>
- 24 Dans certaines disciplines, et notamment l'archéologie, la problématique des relations entre historiens et informaticiens est bien présente depuis trente ans. Les difficultés rencontrées sont souvent liées à la formation initiale des chercheurs, qui n'ont pas appris à programmer et doivent, quand le besoin s'en fait sentir, s'y auto-former. Mais alors se pose la question de la définition de la limite jusqu'à laquelle l'historien doit s'investir dans la programmation, du dialogue avec les informaticiens, de la disponibilité durable dans un projet ou une équipe de recherche des compétences en programmation et de la façon dont ces compétences sont mobilisées. Mais la conviction que le chercheur est le seul à pouvoir comprendre le questionnement et la crainte de l'appauvrissement par l'explicitation l'emporte.

### III. Interrogations

- 25 En premier lieu, le monde des informaticiens et le monde des sciences humaines et sociales ont-ils quelque chose en commun ? La question de la culture générale et de la connaissance de l'autre est à double sens. L'informaticien doit, autant que l'historien, chercher à comprendre le monde de l'autre. Un espace commun doit être créé.
- 26 On peut rappeler que l'adoption de Sparql, XML et R reste une exception chez les historiens. Apprendre à des doctorants à utiliser des outils basiques (traitement de texte, Zotero, etc.) est déjà difficile pour beaucoup de chercheurs. Peut-être faudrait-il qu'un historien-bidouilleur émerge, plutôt qu'un historien-programmeur. Le chercheur pourra ainsi être autonome (et rapide) sur des petites tâches de recherche, tout en évitant l'écueil, pour les grands projets, de faire de mauvais choix techniques, qui, plus tard, risquent d'entraver le partage et la diffusion de ces bases.
- 27 Une historienne estime que, si elle fait de l'informatique, c'est pour résoudre des problèmes qui se posent à elle. Elle travaille alors avec des informaticiens à l'INRIA. La moindre des choses est qu'il y ait une culture partagée : les informaticiens doivent connaître les sciences humaines et sociales, la culture des chercheurs avec qui elle travaille. Quant à la culture informatique des chercheurs, elle doit venir de loin, dès la maternelle. Que veut dire programmer ? Il faut, avant tout, collaborer, coopérer et travailler !

### Conclusion

- 28 Le déficit de formation en informatique des chercheurs en sciences humaines et sociales est flagrant. Leur formation est nettement insuffisante. Elle peut inclure de la programmation, pas tant pour coder que pour apprendre les logiques informatiques.
- 29 Sur les modalités de cette formation, une interrogation subsiste ; l'existence d'une interface évolutive et accessible semble indispensable. Une formation spécifique risque de s'institutionnaliser et donc de se figer dans des certitudes qui ne sont pas du domaine des humanités numériques.
- 30 Se trouve également posée la question de la définition des humanités numériques : discipline ou ensemble de méthodes portant sur des usages nouveaux à la fois de sources et d'outils ?
- 31 Des rencontres régulières à la façon des universités d'été seraient peut-être la meilleure formule pour répondre à ce déficit de formation.
- 32 Les métiers d'historiens et d'informaticiens restent bien distincts. Au lieu de chercher à faire le travail de l'autre, émerge plutôt la nécessité de travailler ensemble, ce qui nécessite une culture commune.
- 33 Les données que produisent les chercheurs dictent, d'une certaine manière, le rôle de l'historien dans la chaîne de développement qui doit expliciter ses méthodes, dès le départ, identifier les risques de cette explicitation liés à la difficulté de « tout » expliciter en amont.

- 34 Il faut, aujourd'hui, pour comprendre la distinction entre le travail du chercheur et celui de l'informaticien, faire la différence entre la structuration des données – que les historiens doivent apprendre – et leur traitement.
- 35 De plus, avant de s'engager dans le développement d'un outil informatique, pour répondre au besoin de traitement de données historiques (ou autres), il est nécessaire de faire l'inventaire de ce qui existe dans son labo, sur le Web, etc., afin d'identifier les outils génériques pouvant être utilisés pour d'autres projets que celui pour lequel ils ont été développés (modèle de données non-spécifiques, usage de normes), d'identifier les outils adaptables pouvant être aisément modifiés (ex : code ouvert, licence de distribution, accès au programmeur) et afin de garantir un bon accès aux données.
- 36 La question de l'historien programmeur a finalement soulevé celle de la formation et de l'articulation du travail entre historien et informaticien.
- 

## NOTES

1. Outre le fameux article d'Emmanuel Le Roy Ladurie, « La fin des érudits », *Le Nouvel Observateur*, 8 mai 1968, qui contient la fameuse citation « L'historien de demain sera programmeur ou ne sera plus », vous pourrez lire les éléments suivants: <http://lists.digitalhumanities.org/pipermail/humanist/2010-October/001666.html>, <http://www.clavert.net/wordpress/?p=385> et <http://www.boiteaoutils.info/2011/09/les-historiens-seront-ils-finalement.html>

2. Pour la première édition : William J. Turkel and Alan MacEachern, *The Programming Historian*, 1st ed. NiCHE: Network in Canadian History & Environment (2007-11), <http://niche-canada.org/programming-historian> ; pour la seconde édition : William J. Turkel (dir.), *The Programming Historian*, <http://programminghistorian.org/>, 2nd ed. NiCHE: Network in Canadian History & Environment (2012-07).

3. Frédéric Clavert fait un essai pour appliquer cette méthode à un domaine précis : en parcourant les sites web de l'Union Européenne ([europa.eu](http://europa.eu) / [ecb.eu](http://ecb.eu)) et de la banque des règlements internationaux ([bis.org](http://bis.org)) avec des mots clés bien réfléchis, le but est de récupérer tout document concernant l'Union économique et monétaire, puis de les stocker dans Zotero et, enfin, de commencer l'analyse avec l'extension (encore instable) pour Zotero appelée *Paper Machines* (<https://github.com/chrisjr/papermachines>). *Paper Machines* autorise une première visualisation, sans avoir besoin de développement, justement.

4. *The Programming Historian* est un guide d'autoformation avec des cours dont la durée varie entre une demi-heure et une heure. Les premiers articles sont des versions remaniées de l'édition de 2007, de nouvelles leçons ont été ajoutées (exemple, la commande Unix *wget*) d'autres sont en préparation (notamment sur Zotero).

5. <http://www.oldbaileyonline.org/>

6. <http://criminalintent.org/>

7. Intervention de F. Beretta et P. Vernus à TCP 2010 sur la modélisation des données en histoire : <http://tcp.hypotheses.org/341> - SyMoGih - [http://larhra.ish-lyon.cnrs.fr/Pole\\_Methodes/SyMoGIH\\_fr.php](http://larhra.ish-lyon.cnrs.fr/Pole_Methodes/SyMoGIH_fr.php)
  8. Graphoskop : <http://www.palaeographia.org/graphoskop/index.htm>
  9. Jean-Philippe Genet, « Histoire, Informatique, Mesure », *Histoire & Mesure* 1 (1986) : 7-18.
  10. Voir les autres chapitres sur la Formations et humanités numériques en France et Quelles compétences et littératies pour les humanités numériques ?
- 

## RÉSUMÉS

À partir du projet de site internet et de livre qui sont devenus collaboratifs pour la préparation d'une deuxième édition - et sur la base d'exemples concrets, cet atelier se propose de discuter de l'intérêt, pour des chercheurs en sciences humaines et sociales, d'apprendre les bases de la programmation.

## INDEX

**Mots-clés :** histoire, programmation, livre collaboratif, sciences humaines